REMARKS

The Office Action dated April 25, 2006 has been received and carefully noted. The following remarks are submitted as a full and complete response thereto.

Claims 1-3, 5-12, 14-23, 25-35, 37-38, and 40-42 stand rejected and pending and under consideration.

REJECTION UNDER 35 U.S.C. § 103:

In the Office Action, at page 2, claims 1-3, 6-12, 14-23, 25-35, 37-38 and 40-42 were rejected under 35 U.S.C. § 103 as being unpatentable over U. S. Patent No. 6,246,680 to Muller et al. ("Muller") in view of U.S. Patent No. 6,999,452 to Murray et al. ("Murray"). The Office Action took the position that Muller and Murray describe all the recitations of independent claims 1, 14, 15, 16, 20, 25, 26, 27, 31, 35, 37, 40, and 41 and related dependent claims. It is respectfully asserted that, for at least the reasons provided herein below, Muller and Murray fail to teach or suggest the recitations of the pending claims. Reconsideration is requested.

Independent claim 1, upon which claims 2, 3, and 5-12 are dependent, recites a network switch, the network switch comprising at least one data port interface supporting a plurality of data ports, a submodule adding an interstack tag into data to keep track of a stack count to prevent looping of the data, at least one stack link interface comprising a bi-directional gigabit stack link interface configured to transmit the data between the network switch and other network switches to create a predetermined configuration, and

a CPU interface, the CPU interface configured to communicate with a CPU. A memory management unit is in communication with the at least one data port interface and the at least one stack link interface, and a memory interface is in communication with the at least one data port interface and the at least one stack link interface, wherein the memory interface is configured to communicate with a memory. A communication channel is provided for communicating data and messaging information between the at least one data port interface, the at least one stack link interface, the memory interface, and the memory management unit. The memory management unit is configured to route data received from each of the at least one data port interface and the at least one stack link interface to the memory interface.

Independent claim 14, upon which claims 21-23 are dependent, recites a scalable network switch, the scalable network switch comprising a predetermined number of switch building blocks interconnected in a meshed configuration. At least one of the predetermined number of switch building blocks comprises at least one data port interface supporting a plurality of data ports for transmitting and receiving data, a submodule adding an interstack tag into the data to keep track of a stack count to prevent looping of the data, a predetermined number of stack link interfaces comprising bi-directional gigabit stack link interfaces configured to transmit the data between one of the predetermined number of building blocks and another of the predetermined number of building blocks to create a predetermined configuration, and a CPU interface configured to communicate with a CPU. A memory management unit is in communication with the

at least one data port interface and the predetermined number of stack link interfaces, a memory interface is in communication with the at least one data port interface and the predetermined number of stack link interfaces, wherein the memory interface is configured to communicate with a memory, and a communication channel is provided for communicating data and messaging information between the at least one data port interface, the predetermined number of stack link interfaces, the memory interface, and the memory management unit.

Claim 15 recites a scalable network switch that includes a predetermined number of switch building blocks interconnected in a meshed configuration. At least one of the predetermined number of switch buildings blocks includes at least one data port interface supporting a plurality of data ports for transmitting and receiving data and a submodule adding an interstack tag into the data to keep track of a stack count to prevent looping of the data. The at least one of the predetermined number of switch building blocks also includes a predetermined number of stack link interfaces comprising bi-directional gigabit stack link interfaces configured to transmit the data between one of the predetermined number of building blocks and another of the predetermined number of building blocks to create a predetermined configuration. Also, the predetermined number of stack link interfaces is configured to be one less than the predetermined number of switch building blocks.

Claim 16 recites a scalable network switch that includes a predetermined number of switch building blocks interconnected in a meshed configuration. At least one of the

predetermined number of switch building blocks includes at least one data port interface supporting a plurality of data ports for transmitting and receiving data and a submodule adding an interstack tag into the data to keep track of a stack count to prevent looping of the data. The at least one of the predetermined number of switch building blocks also includes a predetermined number of stack link interfaces comprising bi-directional gigabit stack link interfaces configured to transmit the data between one of the predetermined number of building blocks and another of the predetermined number of building blocks to create a predetermined configuration. Also, the at least one data port interface includes at least one first data port interface supporting a plurality of first data ports transmitting and receiving data at a first data rate and at least one second data port interface supporting at least one second data port transmitting and receiving data at a second rate.

Claim 20 recites a scalable network switch, the scalable network switch comprising a predetermined number of switch building blocks interconnected in a meshed configuration. At least one of the predetermined number of switch building blocks comprises at least one data port interface supporting a plurality of data ports for transmitting and receiving data, a submodule adding an interstack tag into the data to keep track of a stack count to prevent looping of the data, and a predetermined number of stack link interfaces comprising bi-directional gigabit stack link interfaces configured to transmit the data between one of the predetermined number of building blocks and another of the predetermined number of building blocks to create a predetermined

configuration, wherein each of the predetermined number of stack link interfaces further comprise a gigabit stack link interface configured to transmit and receive data from another gigabit stack link interface on another switch building block.

Claim 25 recites a scalable network switch, the scalable network switch comprising a predetermined number of switch building blocks interconnected in a meshed configuration. At least one of the predetermined number of switch building blocks includes at least one data port interface supporting a plurality of data ports for transmitting and receiving data, a submodule adding an interstack tag into the data to keep track of a stack count to prevent looping of the data, and a predetermined number of stack link interfaces comprising bi-directional gigabit stack link interfaces configured to transmit the data between one of the predetermined number of building blocks and another of the predetermined number of building blocks to create a predetermined configuration. At least one of the predetermined number of switch building blocks further includes a CPU interface configured to communicate with a CPU, a memory management unit in communication with the at least one data port interface and the predetermined number of stack link interfaces, a memory interface in communication with the at least one data port interface and the predetermined number of stack link interfaces, wherein the memory interface is configured to communicate with a memory, and a communication channel, the communication channel for communicating data and messaging information between the at least one data port interface, the predetermined number of stack link interfaces, the memory interface, and the memory management unit.

Claim 26 recites a scalable network switch, the scalable network switch comprising a predetermined number of switch building blocks interconnected in a meshed configuration. At least one of the predetermined number of switch building blocks includes at least one data port interface supporting a plurality of data ports for transmitting and receiving data, a submodule adding an interstack tag into the data to keep track of a stack count to prevent looping of the data, and a predetermined number of stack link interfaces comprising bi-directional gigabit stack link interfaces configured to transmit the data between one of the predetermined number of building blocks and another of the predetermined number of building blocks to create a predetermined configuration, wherein the predetermined number of stack link interfaces is configured to be one less than the predetermined number of switch building blocks.

Claim 27 recites a scalable network switch, the scalable network switch comprising a predetermined number of switch building blocks interconnected in a meshed configuration. At least one of the predetermined number of switch building blocks comprises: at least one data port interface supporting a plurality of data ports for transmitting and receiving data, a submodule adding an interstack tag into the data to keep track of a stack count to prevent looping of the data, and a predetermined number of stack link interfaces comprising bi-directional gigabit stack link interfaces configured to transmit the data between one of the predetermined number of building blocks and another of the predetermined number of building blocks to create a predetermined configuration. The at least one data port interface further comprises: at least one first

data port interface supporting a plurality of first data ports transmitting and receiving data at a first data rate, and at least one second data port interface supporting at least one second data port transmitting and receiving data at a second rate.

Claim 31 recites a scalable network switch, the scalable network switch comprising a predetermined number of switch building blocks interconnected in a meshed configuration. At least one of the predetermined number of switch building blocks includes at least one data port interface supporting a plurality of data ports for transmitting and receiving data, a submodule adding an interstack tag into the data to keep track of a stack count to prevent looping of the data, and a predetermined number of stack link interfaces comprising bi-directional gigabit stack link interfaces configured to transmit the data between one of the predetermined number of building blocks and another of the predetermined number of building blocks to create a predetermined configuration, wherein each of the predetermined number of stack link interfaces further comprise a gigabit stack link interface configured to transmit and receive data from another gigabit stack link interface on another building block.

Claim 35 recites a scalable network switch, the scalable network switch comprising a predetermined number of switch building blocks interconnected in a meshed configuration. At least one of the predetermined number of switch building blocks includes at least one data port interface supporting a plurality of data ports for transmitting and receiving data, a submodule adding an interstack tag into the data to keep track of a stack count to prevent looping of the data, and a predetermined number of

stack link interfaces comprising bi-directional gigabit stack link interfaces configured to transmit the data between one of the predetermined number of building blocks and another of the predetermined number of building blocks to create a predetermined configuration, the scalable network switch further comprising a physical layer transceiver in connection with at least one of the plurality of data ports.

Claim 37 recites a method of stacking network switches. The method includes providing a plurality of clustered switch blocks, and interconnecting each one of the plurality of clustered switch blocks to another one of the plurality of clustered switch blocks. Interconnection of the plurality of clustered switch blocks forms a stack of clustered switch blocks, wherein the step of providing a plurality of clustered switch blocks further includes providing a predetermined number of switch building blocks, interconnecting each of the predetermined number of switch building blocks to every other one of the predetermined number of switch building blocks in a meshed configuration, and adding an interstack tag into data received to keep track of a stack count to prevent looping of the data. Each of the predetermined number of switch building blocks is interconnected to every other one of the predetermined number of switch blocks through an individual stack link.

Claim 40 recites a method of stacking network switches, the method comprising the steps of: providing a plurality of clustered switch blocks, and interconnecting each one of the plurality of clustered switch blocks to another one of the plurality of clustered switch blocks. Interconnection of the plurality of clustered switch blocks forms a stack

of clustered switch blocks. The receiving step further comprises the steps of: receiving a packet on at least one of a data port interface and a stack link interface, adding an interstack tag into the packet to keep track of a stack count to prevent looping of the packet, and storing the packet in a memory in accordance with a predetermined algorithm by allocating memory locations in an internal memory and in an external memory based upon an amount of internal memory available for an egress port of the clustered network switch from which the packet is to be transmitted.

Claim 41 recites a method of stacking network switches, the method comprising the steps of: providing a plurality of clustered switch blocks, and interconnecting each one of the plurality of clustered switch blocks to another one of the plurality of clustered switch blocks. Interconnection of the plurality of clustered switch blocks forms a stack of clustered switch blocks. The forwarding step further includes receiving a packet, adding an interstack tag into the packet to keep track of a stack count to prevent looping of the packet, determining if the destination address of the packet corresponds to a port in the clustered network switch, forwarding the packet to the port corresponding to the destination address if the destination address is determined to correspond to a port in the clustered network switch, determining if the destination address of the packet corresponds to a port on another clustered network switch across a stack, forwarding the packet to a stack link if the destination address is determined to correspond to a port on the another clustered network switch across the stack, and transmitting the packet across

the stack to the another clustered network switch if the destination address of the packet corresponds to a port on the another clustered network switch across the stack.

As will be discussed below, Muller and Murray fail to disclose or suggest the elements of any of the presently pending claims.

According to the bottom portion of page 3 and top portion of page 4 of the Office Action, Muller describes "wherein said memory management unit is configured to route data received <u>from each of said at least one data port interface</u> **<u>and</u>** said <u>at least one stack link interface</u> to the memory interface," as recited in independent claim 1. The Office Action refers to column 5, lines 25-30 of Muller as describing such recitation. However, the referred portion of Muller is limited to describing that "The shared memory manager 220 interfaces with every input port and output port and performs dynamic memory allocation and deallocation on their behalf, respectively. During input packet processing, one or more buffers are allocated in the external shared memory 230 and an incoming packet is stored by the shared memory manager 220 responsive to commands received from the network interface 205, for example." The mere allocation dynamic memory, allocation of buffers, and storing of a packet by the shared memory manager does not teach or suggest that the shared memory manager 220 of Muller routes data from at least one data port interface and at least one stack link interface.

As illustrated in FIG. 2 and described in columns 4 and 5 of Muller, output processing including requesting packet data from the shared memory manager 220 and requesting buffer pointers from the shared memory manager 220. The switch element of

Muller provides that each port of the network interface 205 and the cascading interface 225 includes an input packet process (IPP), an output packet process (OPP), and a media access controller (MAC). See column 5, lines 43-57. The IPPs are coupled in communication with the switch fabric 210, the shared memory manager 220, and the OPPs. The IPPs request forwarding decisions from the switch fabric 210 for received packets and temporarily store the packet data in the shared memory 230 until a forwarding decision is returned. Upon receipt of a forwarding decision, the IPPs forward the corresponding packet to the appropriate OPPs, if any. However, nothing in Muller teaches or suggests that shared memory manager 220 is configured to route data received from each of said at least one data port interface **and** said at least one stack link interface to the memory interface.

Also, the stack link interface of the present application is able to handle a special type of data structure in which items are removed in the reverse order from that in which they are added, so the most recently added item is the first one removed. However, the shared memory manager 220 of Muller is not configured to route data received from at least one stack link interface and a data port interface.

Furthermore, the Office Action correctly recognized that Muller fails to teach or suggest, "a submodule adding an interstack tag into data to keep track of a stack count to prevent looping of the data," as recited in independent claim 1. Accordingly, the Office Action relied on Murray as curing the deficiencies of Muller.

Murray generally describes a packet-switched network system including a multiplicity of multi-port network units having first and second ports and transmission links to support duplex transmission of Ethernet data packets. Each unit transmits from the first and second ports data including selected information enabling on reception of a packet at any of the units a determination of a number of hops from unit to unit around said ring said packet has made. Each unit has a forwarding database and in response to the said selected information controls the transmission of said packets in two directions around.

The Office Action refers to column 7, lines 37-40, as describing, "a submodule adding an interstack tag into data to keep track of a stack count to prevent looping of the data," as recited in independent claim 1. The referred portion of Murray simply describes that if a packet is received on ring port 1 and the number of hops indicated or computed for that packet is equal to the perimeter of the ring, the packet must be discarded. This has no counterpart in 802. 1D functionality. However, Murray does not teach or suggest that a submodule adds an interstack tag into the data. Instead, Murray clearly describes that the number of hops are computed for a particular packet. There is no teaching or suggestion in Murray of addition of such interstack tag as recited in independent claim 1. Thus, clearly and contrary to the contentions made in the Office Action, Murray does not cure the deficiencies of Muller. A combination of Muller and Murray would be devoid of any teaching or suggestion providing the addition of an "interstack tag into data to keep track of a stack count to prevent looping of the data," as recited in independent claim 1.

Accordingly, Muller and Murray fail to teach or suggest all the recitations of independent claim 1 and related dependent claims.

Moreover, the combination of Muller and Murray is improper hindsight combination. The Office Action begins with the template of independent claim 1, for instance, and tries to reconstruct the invention within that template. To protect against such invalid and inappropriate hindsight reconstruction, the Federal Circuit has ruled that references cannot be selected, and selected elements from selected references cannot be combined, without some suggestion, motivation, or teaching that would render obvious that selection and that combination. *See, e.g., Karsten Mfg. Corp. v. Cleveland Golf Co.*, 242 F.3d 1376, 1385, 58 USPQ2d 1286, 1293 (Fed. Cir. 2001) ("In holding an invention obvious in view of a combination of references, there must be some suggestion, motivation, or teaching in the prior art that would have led a person of ordinary skill in the art to select the references and combine them in the way that would produce the claimed invention."); and *Brown & Williamson Tobacco Corp. v. Philip Morris Inc.*, 229 F.3d 1120, 1124-25 (Fed. Cir. 2000) ("a showing of a suggestion, teaching, or motivation to combine the prior art references is an 'essential component of an obviousness holding'").

The Office Action asserted that it would have been obvious to combine the references "in order to prevent looping." The Office Action does not provide any citation for this alleged motivation, and the alleged motivation does not come from the cited references. Nowhere in Murray is there any teaching or any suggestion providing a need

to prevent looping in the architecture described therein. Applicants, thus, respectfully traverse the Office Action's assertion of motivation to combine the references, because it is not based on evidence. The only basis of record for producing what is claimed in independent claim 1 is the present application. Using the present application as the basis for combination, however, is improper hindsight reconstruction.

Because independent claims 14-16, 20, 25-27, 31, 35, 37, 40, and 41 recite similar claim features as those recited in independent claim 1, although of different scope, and because the Office Action refers to similar portions of the cited references to reject independent claims 14-16, 20, 25-27, 31, 35, 37, 40, and 41, the arguments presented above supporting the patentability of independent claim 1 are incorporated herein to support the patentability of independent claims 14-16, 20, 25-27, 31, 35, 37, 40, and 41.

Accordingly, for the foregoing reasons, it is respectfully requested that the rejection of claims 1-3, 6-12, 14-23, 25-35, 37-38 and 40-42 be withdrawn.

<u>REJECTION UNDER 35 U.S.C. § 103:</u>

In the Office Action, at page 11, claim 5 was rejected under 35 U.S.C. § 103 as being unpatentable over Muller, Murray, and U.S. Patent No. 6,775,290 to Merchant et al. ("Merchant"). The Office Action took the position that Muller, Murray, and Merchant discloses all the aspects of dependent claim 5. The rejection is traversed and reconsideration is requested.

As will be discussed below, Muller, Murray, and Merchant fail to disclose or suggest the elements of any of the presently pending claims.

Dependent claim 5 depends from independent claim 1 and recites the additional features of "a variable sized address resolution logic table; and a variable sized VLAN table, wherein said variable sized address resolution logic table and said variable sized VLAN table is in communication with said memory management unit, said at least one stack link interface, and said at least one data port interface." Because the combination of Muller and Merchant must teach, individually or combined, all the recitations of the base claim and any intervening claims of dependent claim 5, the arguments presented above supporting the patentability of independent claim 1 over Muller and Murray are incorporated herein.

Merchant generally describes a method to enable a port of a network switch to support connections with multiple VLANs. See column 1, lines 50-55. Each multiport switch 12 includes a media access control (MAC) module 20 that transmits and receives data packets to and from 10/100 Mb/s physical layer (PHY) transceivers 16 via respective reduced media independent interfaces (RMII) 18 according to IEEE 802.3u protocol. Each multiport switch 12 also includes a gigabit MAC 24 for sending and receiving data packets to and from a gigabit PHY 26 for transmission to the gigabit node 22 via a high speed network medium 28. See column 3, lines 38-47.

However, Merchant does not cure the deficiencies of Muller. Similarly to Muller, Merchant fails to teach or suggest, at least, "a submodule adding an interstack tag into

data to keep track of a stack count to prevent looping of the data," as recited in independent claim 1. Instead, Merchant describes, at most, a frame that may include a VLAN tag header that identifies the frame as information destined to one or more members of a prescribed group of stations. See column 5, lines 45-48. A combination of Muller and Merchant would not provide for the entire claim recitations of independent claim 1, and, accordingly, dependent claim 5.

Accordingly, in view of the foregoing, it is respectfully requested that independent claim 1 and related dependent claim 5 be allowed.

CONCLUSION:

In view of the above, Applicant respectfully submits that the claimed invention recites subject matter which is neither disclosed nor suggested in the cited prior art. Applicant further submits that the subject matter is more than sufficient to render the claimed invention unobvious to a person of skill in the art. Applicant therefore respectfully requests that each of claims 1-3, 5-12, 14-23, 25-35, 37-38, and 40-42 be found allowable and this application passed to issue.

If for any reason the Examiner determines that the application is not now in condition for allowance, it is respectfully requested that the Examiner contact, by telephone, the applicant's undersigned attorney at the indicated telephone number to arrange for an interview to expedite the disposition of this application.

In the event this paper is not being timely filed, the Applicant respectfully petitions for an appropriate extension of time.

In the event this paper is not being timely filed, the applicants respectfully petition for an appropriate extension of time. Any fees for such an extension together with any additional fees may be charged to Counsel's Deposit Account 50-2222.

Respectfully submitted,

Alicia M. Choi
Registration No. 46,621

Customer No. 32294
SQUIRE, SANDERS & DEMPSEY LLP
14^{TH} Floor
8000 Towers Crescent Drive
Tysons Corner, Virginia 22182-2700
Telephone: 703-720-7800
Fax: 703-720-7802

AMC:jkm